ELSEVIER
Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia Engineering 29 (2012) 3636 – 3640
Procedia Engineering
www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

Dynamic Routing Algorithm Based on Mobile Agent

Li Lan[a*]

[a]*College of Computer and Information Engineering，Harbin University of Commerce，Harbin, China*

Abstract

Routing is briefly explained in this paper, as well as the mechanism and the shortcomings of current dynamic routing are analyzed. After briefly introducing the advantages of mobile agent, the paper focuses on applying the new Mobile Agent technology to routing algorithm and proposes an innovative dynamic routing algorithm based on MA.

Keywords: Dynamic Routing; Mobile Agent; Routing Algorithm

1. Introduction

Contemporary computer networks are heterogeneous; even a single network consists of many kinds of processors and communications channels. As connections proliferate, network topologies necessarily become more and more dynamic. Devices may move from place to place, or maintain intermittent connections, or change their relationships to the network and their peers on the fly. Networks must be flexible enough to allow these devices to communicate with each other in a variety of ways and across a variety of substrates.

Dynamic routing refers to the automatic adjustment process of Routers in accordance with the network topology and communication flows [1], which is applicable in current networks with a large scale, a large amount of switching nodes and high traffic. Dynamic routing uses a dynamic routing protocol to automatically select the best route to put into the routing table. The so-called routing refers to confirming a channel that connects the source node and the destination node in the network by choosing the intermediate nodes of message switching and links between nodes. Routing algorithm can be divided into

* Corresponding author. Tel.: 0451-84892118.
E-mail address: lilan9339cn@sina.com.

doi:10.1016/j.proeng.2012.01.544

link state algorithm and distance vector algorithm in accordance with different routing algorithm discoveries and methods to calculate route [2].

An agent is usually defined as an independent software program that runs on behalf of a network user. It can be characterized as having more or less intelligence and it has the ability to learn. Mobile agents (MA) add to regular agents the capability of traveling to multiple locations in the network, by saving their state and restoring it in the new host. Mobile agents are a novel way of building distributed software systems. Traditional distributed systems are built out of stationary programs that pass data back and forth across a network. Mobile agents, by contrast, are programs that they move from node to node: the computation moves, not just the attendant data. The mobile agent system includes two parts: the agent operation environment and the agent [3]. Most systems which adopt the MA technology can be realized by using current technology. The reason of using MA is that comparing with traditional method, its design, realization and execution has irreplaceable advantages: high-efficiency, reducing burden on the network, providing real-time remote interaction, supporting off-line computation, asynchronous autonomous interaction, dynamic adaptability, easy for distribution, with heterogeneous environment, personalized server behavior and convenient development mode [4, 5]. It must be pointed out that the advantage mentioned above is not unique to MA technology. Except for remote real-time interaction, other advantages can also be realized by using traditional methods, but only the MA technology can provide an architecture framework that can satisfy all requirements.

2. Dynamic Routing Algorithm Based on Mobile Agent

2.1. Description of Routing Nodes

Routing nodes are responsible to forward user data and ultimately deliver it to the destination of data packet. In addition, routing nodes are also the source, intermediate nodes and destination of routing MA. Routing MA moves between the routing nodes and realizes information interaction between agents by reading-writing two specific data tables on the network nodes. Each node is responsible to maintain and update the two data tables: the routing table and the distance statistical table. The routing table T_k includes selection probability value P_{nd} of different adjacent nodes aiming at different destination nodes, which refers to the probability of choosing the adjacent node n as the next routing node when the destination node is d. In accordance with different destination nodes of user data, routing nodes will choose its next routing node based on this value. Different from traditional routing table, the principles of flow equilibrium can be comprehensively considered, and data can be forwarded to each node in accordance with the selection probability ratio between various adjacent nodes. In this way, the network resource can be adequately utilized and burden on a single link can be reduced. Unit data of the routing table should satisfy Eq. (1), i.e. for all adjacent nodes with the same destination node, the sum of their selection probabilities should be 1.

$$\sum_{n\in N_k} P_{nd} = 1,\ d \in [0, N] \tag{1}$$

Of which, P_{nd} refers to the probability of choosing the adjacent node n as the next routing node when the destination node is d; N_i refers to the number of adjacent nodes of node d; N refers to the node number of current network. The data table $M_k(\mu_d, \sigma_d^2, \overline{\omega}_d)$ defines a simple parameter statistical model of various nodal distances in the whole network observed from node k. This model is self-adaptive. It updates and gets the mean μ_d and variance σ_d^2 of the statistical model in accordance with the feedback of MA's travel time, and uses $\overline{\omega}_d$ to save the optimum agent travel time in the window during the latest time.

2.2. Elaboration of the Algorithm

On each routing node s, a routing MA with d as the destination node is generated for every interval Δ t, and it is used to find a feasible and low-cost route of s->d, and in the meantime examine the load condition of the network. With each passed node, the mobile agent and user data enjoy the same priority and experience the same congestion situation, thus can make accurate evaluation of the network environment. Selection of the destination node d is in accordance with the generation mode of the data traffic load of this node. The probability to generate a routing agent on node s with d as the destination Pd is Eq. (2).

$$P_d = f_{sd} \Big/ \sum_{d'=1}^{N} f_{sd'}\,,\quad N = \{neighbours(s)\} \tag{2}$$

Of which, s refers to the routing node which generates the mobile agent; d refers to the destination node of the mobile agent; pd refers to probability to generate a mobile agent on node s with d as the destination; f_{sd} refers to the measured value of the data sent from node s to node d; $f_{sd'}$ refers to the measured value of the data sent from node s to node d'. During the process of moving toward the destination node d, the routing agent will save the routing node identifier it has passed by and the traffic situation it has found out. Identifier of each visited node k and the time between generation of agent and sending of node k is pushed in to the stack of MA $s_{s\text{->}d(k)}$, and moves to the next node with MA until reaching the destination node d. On each intermediate node k, MA visits the routing table (it can only visit). Choose one node n as the next forwarding node from the adjacent nodes which have never been visited before and have node d as the destination node, and the selection range of the next node includes all adjacent nodes. The selection parameter P'_{nd} will consider the routing selection probability P_{sd} of the routing table, combining the situation of the nth link queue on node k, calculate in accordance with Eq. (3), and we have:

$$P'_{nd} = (P_{nd} + \alpha l_n) \big/ [1 + \alpha(|N_k| - 1)]\,,\ ln \in [0, 1] \tag{3}$$

Of which, k refers to the current routing node where routing agent is; l_n refers to heuristic correction coefficient, which is proportional to the buffer queue length of the adjacent node n of current node k, and it reflects the real-time state of node buffer zone and is related to the queue waiting time, see Eq. (4); α refers to the correction degree of l_n toward P_{nd}; $|N_k|$ refers to the number of adjacent nodes of current node k; P'_{nd} refers to the selection parameter with d as the destination node and n as the next forwarding node.

$$l_n = 1 - q_n \Big/ \sum_{n'=1}^{|N_k|} q_{n'} \tag{4}$$

Of which, q_n refers to the number of packets sent to the adjacent node n in the buffer zone of current node; $q_{n'}$ refers to the number of packets sent to the adjacent node n' in the buffer zone of current node.

The routing selection probability in the routing table has continuously studied past and current situations of the whole network, so using heuristic correction coefficient to conduct correction will make the algorithm have more feedback. α is used to measure the importance of heuristic correction coefficient compared with the probabilities saved in the routing table. From the experiment we find that if α is between 0.2 and 0.5, the algorithm performance will not have obvious change.

After reaching the destination node d, MA will process the stack, then sent update information to service agent of each node it has passed by during the route in accordance with the stack data, and then this MA dies. The sent information data packet includes the latest data used to update various node routing tables and traffic model tables. After the node service agent has received the data, it will update all items related to the destination node d in the two main data units of the node in accordance with the data.

2.3. Update of the Traffic Statistical Model

This algorithm conducts update to model data in accordance with Eq. (5).

$$\begin{cases} \mu_d \leftarrow o_{k\to d} \\ \sigma_d^2 \leftarrow 0 \end{cases}, \quad other \begin{cases} \mu_d \leftarrow \mu_d + \eta(o_{k\to d} - \mu_d) \\ \sigma_d^2 \leftarrow \sigma_d^2 + \eta((o_{k\to d} - \mu_d)^2 - \sigma_d^2) \end{cases} \tag{5}$$

Of which, k refers to current routing node; d refers to the destination routing node; $O_{k\text{->}d}$ refers to the latest observed travel time of routing MA from node k to the destination d; η refers to the mobile index window, which limits the impact of past travel time of routing agent on the calculation of μ, so that the statistical model can more accurately reflect the long-term network situation. Mobile observation window $\overline{\omega}_d$ is used to calculate the minimum travel time $\overline{\omega}_{best-d}$ of the nearest w routing MAs which have passed by node k and have d as the destination node. $\overline{\omega}_{best-d}$ represents the lower limit of the travel time from node k to node d during the past period of time, which reflects the short-term network situation.

2.4. Update of the Routing Table

Update of the routing table is in accordance with the data of the stack unit, which uses the travel time T from current node k to the destination node d to update the mean μ_d, variance σ_d^2 and optimum value $\overline{\omega}_d$ in the observation window.

When choosing node n as the next node form node k to the destination node d, the update of the routing table requires increase of the routing selection probability in the routing table and reduction of other units with d as the destination node, but it should satisfy Eq. (1). The size of probability change depends on the evaluation on the travel time, because no matter from the physical perspective or from the perspective of network congestion, the length of travel time is directly proportional to the travel route. However, travel time composed of the time spent by various sub-routes cannot be considered as good criteria to measure if it is good or not, because we don’t know what is good travel time, which should be evaluated in accordance with the load situation of the whole network.

3. Experimental results and analysis

3.1. Experiment scheme

The experimental analogy network is going to adopt one simple network topology with 6 network node, as shown in Fig 1.

Each experimental analog time is set as 300s, and routing agent and user data are sent simultaneously on each node. Before each analog sending of data unit, the experimental system should pre-run for 60s, only forwarding mobile agent and responsible for initializing the routing table of each node. In this experiment, node 1 only sends data with 6 as the destination node in fixed speed, and in the meantime generates mobile agent in accordance with the data flow. Initial routing table should be set as that, for the same destination node, its adjacent nodes should have the same selection probability; initial values of the traffic statistical model μ, σ, ω are all set as 0.

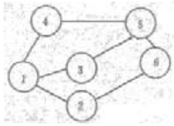

Fig. 1. Experimental network topology schemes

3.2. Experimental results and data analysis

Record of the User Data: In the experiment, set node 1 as the source node, and it generates user data to node 6 on a regular basis. After the experiment, statistics of user data received and forwarded by each node is shown in Fig 2 (a).

Record of the Routing Data: In the experiment, routing mobile agent with 6 as the destination node is sent from node 1, and the sending interval is set as 2s. When the experiment begins, at first, only routing agents are sent to the network to conduct initialization of the node routing table; then, during the process of sending user data, routing mobile agents are also constantly generated to obtain updated routing information. Specific experimental data is shown in the Fig 2 (b).

From the above data, we can see that the best path is from node 1, passing node 2, and then reaching node 6. We can see that this algorithm can well choose the optimum routing path.

4. Conclusions

This paper proposes a dynamic routing algorithm based on mobile agents, bringing in advantages of mobile agents which are applicable in distributed computation, and increasing security; using probabilities to build the routing table is convenient for flow equilibrium; the network topology is not saved in the routing node but actively discovered through mobile agents, which saves resources.

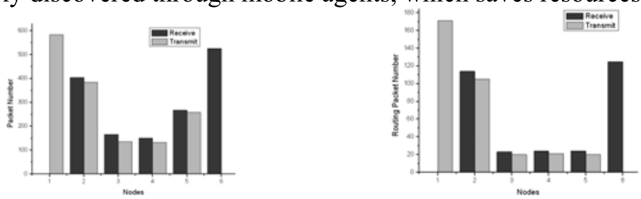

Fig. 2. (a) Routing node user data statistics; (b) Node transceiver routing mobile agent statistics

References

[1] Mark A Sportack. IP Routing Fundamentals. Publishing House of Electronics Industry, 1999

[2] Wang bo, Wang hongman, Zhou hua. Distributed Computing Environment. Beijing University of Posts and Telecommunications Press, 2000

[3] White JE. Mobile agents. In: Bradshaw, Jeffrey eds. Soft-ware Agents, Menlo Pork/ California: AAAI Press/ The MIT Press, 1996

[4] Lange D B. Mobile objects and mobile agents: The future of distributed computing. In: Proc of the European Conf on Object-Oriented Programming' 98, Brussels, 1998

[5] Chess D M, Harrison C G, Kershenbaum A. Mobile agents: Are they a good idea. TechRep: RC1 9887, IBM T JWaston Research Center, 1995